

NASA Technical Memorandum 105572

1N-61
140764
P.43

User's Manual for Interactive Data Display System (IDDS)

James D. Stegeman
Lewis Research Center
Cleveland, Ohio

(NASA-TM-105572) USER'S MANUAL FOR
INTERACTIVE DATA DISPLAY SYSTEM
(IDDS) (NASA) 43 p

N93-16613

Unclass



National Aeronautics and
Space Administration

Lewis Research Center
Cleveland, Ohio 44135

G3/61 0140764



TABLE OF CONTENTS

ORIGINAL DOCUMENT
ON MICROFILM

1.0 Summary	1
2.0 Introduction	1
3.0 Data Input	2
3.1 Introduction	2
3.2 Data and Grid Files	2
3.2.1 Grid files	2
3.2.2 Data files	4
3.2.3 Real-time	5
3.3 Geometry File	5
3.4 Header File	5
4.0 Menu System	6
4.1 Introduction	6
4.2 Getting Started	6
4.3 IDDS Main Menu	7
4.3.1 Input Data menu	7
4.3.2 Experimental data	7
4.3.3 Simulation Data menu	8
4.3.4 Retrieve from data base	8
4.4 Screen Layout Menu	8
4.4.1 Single window	8
4.4.2 Multiple windows	8
4.5 Files and Windows	9
4.6 Display Specifications	9
4.6.1 Variable selection	10

4.6.2	Plane selection	10
4.6.3	Plot type selection	11
4.6.4	Color bar selection	11
4.6.5	Labeling selections	11
4.6.6	Mode selection	12
4.6.7	Display options	12
4.6.7.1	Contour options	12
4.6.7.2	Vector options	12
4.6.7.3	Grid options	12
4.6.7.4	Carpet plot options	12
4.6.7.5	X-Y plot options	12
4.6.7.6	Axis options	12
4.6.7.7	Delay time option	12
4.6.7.8	Geometry option	12
4.6.8	Control options	12
4.7	Geometry Menu	13
4.8	Display Control Menu	14
5.0	Plotting Capabilities	14
5.1	Contour Plots	14
5.2	Vector Plots	15
5.3	Shaded Plots	15
5.4	X-Y Plots	15
5.5	Carpet Plots	15
6.0	Example Session	15
Appendix A	35
Appendix B	43

1.0 SUMMARY

A computer graphics package for the visualization of three-dimensional flow in turbomachinery has been developed and tested. This graphics package, called IDDS (Interactive Data Display System), is able to "unwrap" the volumetric data cone associated with a centrifugal compressor and display the results in an easy to understand two-dimensional manner. IDDS will provide the majority of the visualization and analysis capability for the ICE (Integrated CFD and Experiment) system. This document is intended to serve as a user's manual for IDDS in a stand-alone mode. Currently, IDDS is capable of plotting two- or three-dimensional simulation data, but work is under way to expand IDDS so that experimental data can be accepted, plotted, and compared with a simulation dataset of the actual hardware being tested.

2.0 INTRODUCTION

The Interactive Data Display System (IDDS) has been developed for use in the Integrated CFD and Experiment (ICE) system environment. In the ICE environment, IDDS can be invoked from any subsystem, or it can be used as a stand-alone display package. This first chapter provides an overall view of the operation of the package. The remaining chapters serve as a user's manual for the IDDS graphics package.

IDDS runs on an Iris workstation equipped with at least 16 MB of main memory and having at least 24 color and 24 z-buffer bit planes. Alpha planes are not required. The package itself is menu driven, with selections made primarily through the mouse. There are at most six levels to the menu system. To invoke IDDS in a stand-alone mode, you must select the **Analysis** subsystem and then **Plot System** from the top two menu levels. The menu system is then broken down into four main submenus: **Input Data**, **Screen Layout**, **Display Specifications**, and **Geometry**, which is a submenu of **Display Specifications**.

When you select **Input Data**, you can choose between simulation and experimental data. This distinction is for plot comparison, which is not yet available. Either type of data can be two-dimensional or three-dimensional. A maximum of four data files can be specified through this menu; they can be a mixture of two-dimensional, three-dimensional, simulation, or experimental data.

Associated with each data file are a grid file and optional geometry and header files. Full descriptions of all files will be presented later in this document. Once you have input the file names, you can select to have the data converted to cylindrical coordinates. This conversion process lets you "straighten out" a highly curved geometry (i.e., centrifugal compressor), to get a better look at the data. You are responsible for assuring consistency of units between header information and the input data file.

The **Screen Layout** selection allows you to select the number of windows and, hence, plots displayed on the screen. The maximum number of windows displayable at once is four. Since the maximum number of files is also four, each window can have a unique file associated with it for plot comparison.

Display Specification allows you to customize the plot(s) on the screen. This selection is active only after you have made your **Input Data** and **Screen Layout** selections. In this menu you can select, among others, the variable for display, slice orientation through the data, type of plot, numeric and color ranges for the color bar, contour levels, and vector levels. From this menu, you can also enter the **Geometry** menu. Geometric coordinates are taken directly from the grid file. IDDS was designed to allow the user to display the geometry, orient it as desired, and then overlay the desired plot on top of the geometry.

When you are finished selecting the desired plot(s) and geometry, you must then enter the **Display** menu to "build" the display. Here, you have the option to display the grid, the plot, or the geometry by themselves or in any combination. After the picture is rendered, you can rotate, translate, or zoom on the display.

The remaining chapters of this manual are divided up as follows. Chapter 3.0, Data Input, describes the required format and other input capabilities. Chapter 4.0, Menu System, describes and analyzes the overall menu structure. Chapter 5.0, Plotting Capabilities, deals with the various plot types, their options, and ways to display them by themselves or combine them with the geometry. Chapter 6.0 is an example session at the terminal.

3.0 DATA INPUT

3.1 Introduction

This chapter is intended to serve as a guide for formatting files to be input to IDDS. A total of four files can be associated with one another. Two files are mandatory, the data and grid files, and two files are optional, the geometric description file and the data header file. The data and grid files are required to be in binary, whereas the geometric and header files are standard ASCII files. The remainder of this chapter will consist of file structure descriptions and examples for illustration purposes.

3.2 Data and Grid Files

3.2.1 Grid files. — The file format required by IDDS for both simulation and experimental data and their associated grid files is that of PLOT3D. By taking this approach, existing datasets will not have to be converted to a new format, and codes will not need to have output routines modified. Figure 1 shows a generic two-dimensional grid.

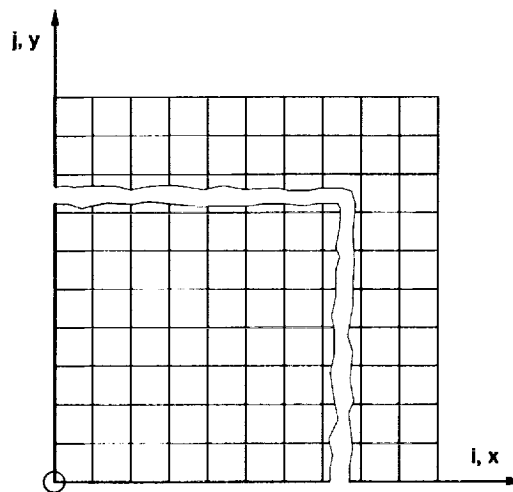


Figure 1.—Generic 2 D grid.

The grid shown in figure 1 has n -columns in the i -direction and m -rows in the j -direction. To write this grid out properly, you must observe the following guidelines:

- (1) Starting at the lower left corner, go across (in i) writing out x -coordinates until all columns are finished.
- (2) Repeat step 1 for the next row until all rows are finished.
- (3) Perform steps 1 and 2 for the y -coordinates.

The following code written in the C programming language will illustrate this better.

```
fwrite(&n_col, 4, 1, grid_file);
fwrite(&n_row, 4, 1, grid_file);

for(j = 0; j < n_row; j++)
    for(i = 0; i < n_col; i++)
        fwrite(&x[i][j], 4, 1, grid_file);
```

```

for(j = 0; j < n_row; j++)
  for(i = 0; i < n_col; i++)
    fwrite(&y[i][j], 4, 1, grid_file);

```

where

```

n_row    = m
n_col    = n
x, y     = Arrays containing grid coordinates
grid_file = Pointer to name of grid file being generated

```

To extend this to a three-dimensional format, simply add the z-coordinates after the y-coordinates. For example, figure 2 illustrates a generic three-dimensional grid with $n_row = 6$, $n_col = 7$, and $n_surf = 4$.

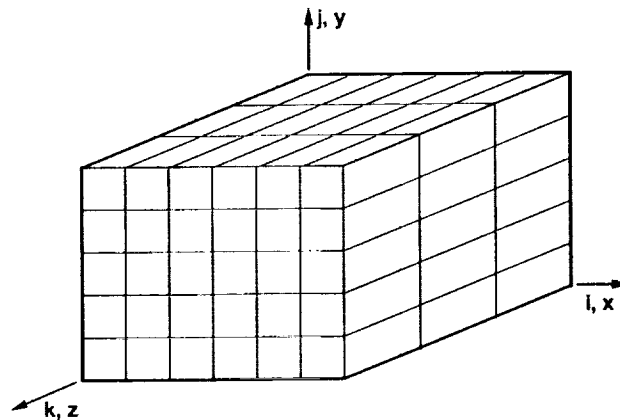


Figure 2.—Generic 3 D grid.

The code to write out this three-dimensional grid follows:

```

fwrite(&n_col, 4, 1, grid_file);
fwrite(&n_row, 4, 1, grid_file);
fwrite(&n_surf, 4, 1, grid_file);

for(k = 0; k < n_surf; k++)
  for(j = 0; j < n_row; j++)
    for(i = 0; i < n_col; i++)
      fwrite(&x[i][j][k], 4, 1, grid_file);

for(k = 0; k < n_surf; k++)
  for(j = 0; j < n_row; j++)
    for(i = 0; i < n_col; i++)
      fwrite(&y[i][j][k], 4, 1, grid_file);

for(k = 0; k < n_surf; k++)
  for(j = 0; j < n_row; j++)
    for(i = 0; i < n_col; i++)
      fwrite(&z[i][j][k], 4, 1, grid_file);

```

where

n_row = 6
 n_col = 7
 n_surf = 4
 x, y, z = Arrays containing grid coordinates
 grid_file = Pointer to name of grid file being generated

3.2.2 Data files. — The discussion in the preceding section on how to break down and write out a grid also applies to the generation of a data file. Each variable is written out as if it were a coordinate. That is, variable 1 is written out for all nodes, and then variable 2 is written out for all nodes, and so on. This continues until all variables are written out for all nodes in the grid.

Four items of required information must precede the information about the variable data. They are, as follows, in order:

- (1) Amach
- (2) Alpha
- (3) Reynolds
- (4) Time

You are responsible for assuring unit consistency between data and parameters. When you are inputting vector data, the first few data slots must be as follows:

<u>Two-dimensional</u>	<u>Three-dimensional</u>
(1) Rho	(1) Rho
(2) Vx	(2) Vx
(3) Vy	(3) Vy
(4) Anything	(4) Vz
	(5) Anything

for rotating machinery, and

<u>Two-dimensional</u>	<u>Three-dimensional</u>
(1) Anything	(1) Anything
(2) Vx	(2) Vx
(3) Vy	(3) Vy
(4) Anything	(4) Vz
	(5) Anything

for nonrotating machinery. Appendix A contains example codes for the generation of two-dimensional and three-dimensional data files.

When rotating machinery data is input to IDDS, you have the option of converting this data to cylindrical coordinates. By converting the coordinates to cylindrical format, highly curved grids are, in effect, “straightened out” or “unwrapped.” Since the algorithm to perform the unwrapping function is predicated upon the i-index corresponding to x, the j-index to y, and the k-index to z (with z being the axis of rotation), there is no guarantee that this function will work properly if these norms are not adhered to. Once the grid is unwrapped, IDDS then transforms the velocity components Vx, Vy, and Vz to new components Vr, V_θ, and Vz, respectively. The cylindrical components are then mapped into the unwrapped space, and are available for display in all views. If you decide to use the convert option, prompts will be issued for the following information:

- (1) Omega (rpm)
- (2) Stagnation temperature
- (3) Total pressure
- (4) C_p

Again, you are responsible for assuring unit consistency between these parameters and the data.

3.2.3 Real-time.—The real-time data acquisition and display portion of IDDS is unavailable at the present time.

3.3 Geometry File

The geometry file allows you to define geometric structures via the grid file by specifying a name for an object and then its index ranges. A maximum of 10 objects can be defined per grid file. As an example, suppose the grid input is of size 45 by 44 by 106 (i, j, k) with the hub at j = 1, the leading edge at k = 26, the trailing edge at k = 86, the blade tip at j = 40, the suction surface at i = 1, and the pressure surface at i = 45. This example geometry is shown in figure 3. With figure 3 in mind, the geometry file would appear as shown in figure 4. Please note that only the information listed beneath the heading Content should be in the geometry file.

The geometric file is a standard ASCII file and can be created with any editor. If no geometry file is specified, then the geometry drawing portion of IDDS is not available. That is, IDDS does not have any facility for reading in these specifications interactively.

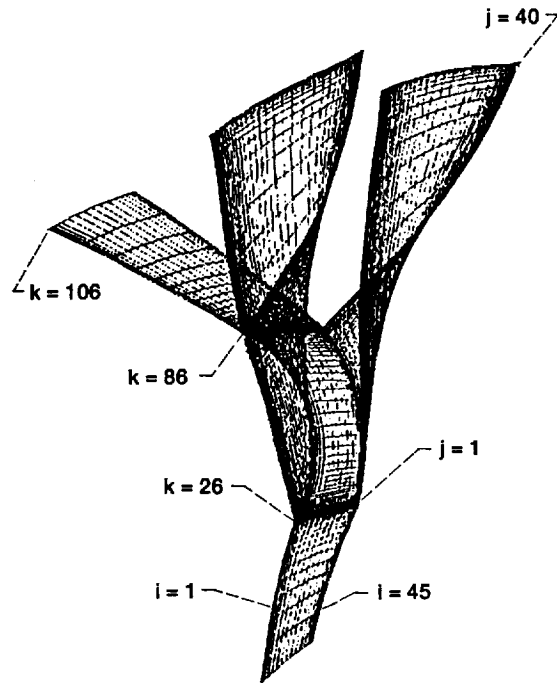


Figure 3.—Example geometry

Line	Content	Description
1	3	Number of objects defined here
2	21	Number of blades in machine
3	HUB	Name of first object
4	1 45 1	i-index range for HUB
5	1 1 1	j-index range for HUB
6	1 106 1	k-index range for HUB
7	PRESSURE	Name of second object
8	1 1 1	i-index range for PRESSURE
9	1 40 1	j-index range for PRESSURE
10	26 86 1	k-index range for PRESSURE
11	SUCTION	Name of third object
12	45 45 1	i-index range for SUCTION
13	1 40 1	j-index range for SUCTION
14	26 86 1	k-index range for SUCTION

Figure 4.—Geometric file example.

3.4 Header File

The header file allows you to input variables in addition to those associated with a standard PLOT3D data file. Specified in this file are the number of variables to be input to IDDS, the names of these variables, and the number of solutions (fig. 5).

Line	Content	Description
1	2	Number of grids
2	7	Number of variables
3	Rho	Name of variable 1
4	Velocity_x	Name of variable 2
5	Velocity_y	Name of variable 3
6	Velocity_z	Name of variable 4
7	Total_Energy	Name of variable 5
8	Pressure	Name of variable 6
9	Enthalpy	Name of variable 7

Figure 5.—Header file example.

This file, like the geometry file, is an ASCII file and can be created with any standard editor. Note that a variable name consisting of two words (Total Energy) or two parts (Velocity x) must be joined by a non-blank character (i.e., an underscore or a hyphen). This is because of the C-language routine used to read the file. IDDS can accommodate a total of 10 variables. If this file is not specified, standard PLOT3D variable names will be used.

The first line in the example header file (fig. 5) indicates that there are two solutions to be processed. These solutions are treated as distinct models of the simulation space at successive points in time and are read from the same file. These solutions are not displayed on the screen simultaneously. Rather, they can be displayed successively. As a result, the size of the individual grids can vary when you are using multiple grids.

As an example, a code which uses a multi-grid technique can output the grid as it proceeds from coarse to fine (or vice versa), and you can see this on the screen. Any data quantity mapped to these grids can then be displayed. Another example of usage is shown in figure 6. The grid in figure 6 stretches and compresses as necessary when the piston is traveling to bottom dead center or to top dead center, respectively. On all grids in this process, a fully converged solution is computed, and it can be stored and displayed using this multi-grid technique.

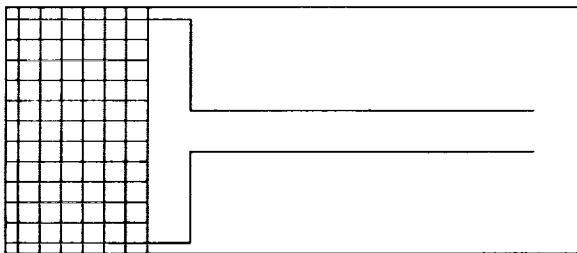


Figure 6.—Piston engine.

If the grid size and coordinates are constant across multiple grids, you must input the first grid and can then follow that with -1's for the successive grid dimensions (i.e., -1 -1 for two-dimensional and -1 -1 -1 for three-dimensional). IDDS interprets this to mean "use the last grid read in." The geometry file can be defaulted in this manner by using a -1 on the first line for the successive descriptions. The solution file cannot be defaulted in this manner. IDDS is currently able to accommodate 150 solutions and their associated grids in this fashion. Appendix A contains various example programs illustrating the previous discussion. If this file is not specified, standard PLOT3D variable names and one solution will be the default.

4.0 MENU SYSTEM

4.1 Introduction

This chapter describes the four major menus of IDDS. First to be covered will be the **Input Data** menu. You can use this menu and its submenus to input the grid and data file names. A maximum of four pairs of files can be manipulated by IDDS. The second menu to be covered is the **Screen Layout** menu. In this menu, you can configure the screen with from one to four windows. Each window can have a unique file associated with it. Next will be the **Display Specifications** menu. The user can customize the display in this menu. The last menu to be discussed will be **Display Control**. In this menu, you can determine what is to be displayed: the grid, the plot, the geometry, or any combination of these.

4.2 Getting Started

To get started, type IDDS <return>. The screen should now appear as in figure 7.

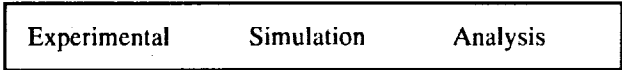


Figure 7.—Global menu.

The **Simulation** and **Experimental** boxes on the screen are for use with future ICE software and are currently not available. To start IDDS, select the **Analysis** box on the screen. To make a selection in this menu system, place the cursor over the desired box and press the left mouse button. To exit from the Global Menu, press the <Esc> key.

After you select the **Analysis** subsystem, the screen will appear as shown in figure 8. From this menu, select **Plot System**. The **Data Base System** choice is not available at this time; **Reset** clears the **Analysis** subsystem; **Help** explains the choices in this menu; and **Return** takes you back to the Global Menu (i.e., back to fig. 7).

Plot System	Data Base System	Reset
Help	Return	

Figure 8.—Analysis subsystem main menu.

4.3 IDDS Main Menu

Selecting **Plot System** takes you to the IDDS main menu. This menu gives access to the first three major menus of IDDS: **Input Data**, **Screen Layout**, and **Display Specifications**. The screen will appear as shown in figure 9.

Input Data	Screen Layout	Display Specifications
Help	Return	

Figure 9.—IDDS main menu.

You must select **Input Data** and **Screen Layout** before you select **Display Specifications**. The order you select **Input Data** and **Screen Layout** does not matter. **Help** provides information on the choices in this menu. The **Return** selection backs the menu system up one level, which, in this case, would be to the menu shown in figure 8. The **Help** and **Return** selections function consistently throughout IDDS and will not be discussed further.

4.3.1 Input Data menu.— When you select **Input Data** from the IDDS main menu, the screen will appear as shown in figure 10. You can input a maximum of four files of any type via this menu.

Now, make a choice from the top row.

Experimental Data	Simulation Data	Retrieve From Data Base
Help	Return	

Figure 10.—Input Data menu.

4.3.2 Experimental data.— Selecting **Experimental Data** from the **Input Data** menu tells IDDS that the file contains experimental data. Although the format is the same for experimental and simulation data, the distinction is made here so that it is properly identified when and if it enters the data base. Also, the file is tagged for future enhancements to IDDS such as file comparisons between experimental and simulation data. The screen will now appear as shown in figure 11.

Select from the four types of experimental data from the top row. Once you make a selection, you are prompted to type in the data file name and the grid file name. If the files are not located in the current directory, the relative or absolute path to the file must precede the file name. Once you enter the file name, press the mouse button again to enter multiple datasets. Adding a file to the data base is an option that is not yet available.

R-theta-Z	X-Y-Z	R-theta	X-Y
Add to Data Base	Help	Return	

Figure 11.—Experimental data menu.

4.3.3 Simulation Data menu.— When you select **Simulation Data**, you are telling IDDS that the file(s) to be input contain simulation data. As mentioned previously, this distinction is for logistical purposes and future enhancements to IDDS. The screen will appear as shown in figure 12.

Now make a selection from the top row of this menu. Once you input the name, you can click the mouse button to enter multiple datasets. Again, adding a file to the data base is an option that is not yet available.

2-D Data	3-D Data	Add to Data Base
Help	Return	

Figure 12.—Simulation Data menu.

4.3.4 Retrieve from data base.— The **Retrieve From Data Base** function listed in the **Input Data** menu (fig. 10) is currently being developed and will be released at a later date.

4.4 Screen Layout Menu

The **Screen Layout** menu system (see fig. 9) allows you to configure the screen with from one to four display windows. There are eight possible screen configurations. Once the screen configuration has been specified without ambiguity, you are prompted to tell IDDS which file to associate with which window. This menu will be discussed after the screen configuration choices are presented. The screen now appears as shown in figure 13.

Single Window	Multiple Windows	Help	Return
---------------	------------------	------	--------

Figure 13.—Screen Layout menu.

4.4.1 Single window.— When you select **Single Window**, you are telling IDDS to use the entire screen for displaying the plots. You will now go to the Window and File Association menu.

4.4.2 Multiple windows.— When you select **Multiple Windows**, you are telling IDDS to divide the screen up for multiple displays. At this point, you are presented with a series of choices to determine the desired screen configuration. The first of these screens is shown in figure 14.

Two Windows	Three Windows	Four Windows
Help	Return	

Figure 14.—Multiple window selection.

If you select **Two Windows**, you must decide between the choices shown in figure 15.

Left-Right	Upper-Lower
------------	-------------

Figure 15.—Two window choices.

If you select **Three Windows**, you must decide between the selections shown in figure 16.

Left-UR-LR	Right-UL-LL	Upper-LL-LR	Lower-UL-UR
------------	-------------	-------------	-------------

Figure 16.—Three window choices.

Here, **UR** indicates “Upper Right Quadrant,” **LL** indicates “Lower Left Quadrant,” and so forth.

With any selection from figure 15 or 16, you have made an unambiguous screen configuration known to IDDS and will then be placed in the Window and File Association menu. Selecting **Four Windows** from figure 14 automatically divides the screen up into four equal quadrants. This is an unambiguous choice. In summarizing, figure 17 shows the possible ways in which the screen can be divided up.

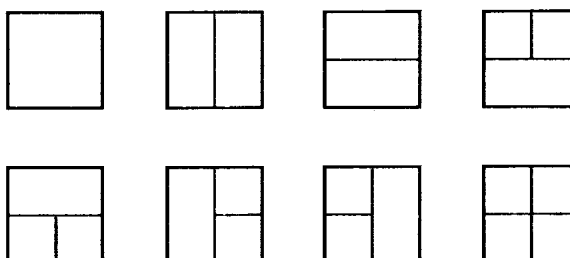


Figure 17.—Screen configurations.

4.5 Files and Windows

When you have input at least one file name and the desired screen configuration to IDDS, you will be prompted to associate a file with a window. For example, if you specify four file names and the screen is configured for two windows, you will see the screen shown in figure 18.

Upper	File 1
Lower	File 2
	File 3
	File 4
Reset Window	Bind
Reset File	Return
Reset All	Help

Figure 18.—Window and File Association menu.

In this example, you must now select a window (upper or lower), and one of the four files. An asterisk will be placed next to the items selected. Once selected, they must be bound together using **Bind**. The file name will now be written to the right of the intended window name. Before you select **Bind**, you can use **Reset File** and **Reset Window** to reset the file(s) selected to nothing and the window(s) selected to nothing, respectively. Once items are bound, these selections do not perform their functions, and **Reset All** is the only way to change screen configurations. Selecting **Return** tells IDDS that the screen configuration is complete. IDDS will now start reading the selected files, scan the data for minimum and maximum values, and, if necessary, perform the conversion to cylindrical coordinates. These processes can take a little while to complete, so please be patient.

4.6 Display Specifications

Selecting **Display Specifications** from the IDDS main menu (fig. 9) allows you to specify all aspects of the desired plot(s), such as variables, planar orientations, plot type, color bar limits and colors, labels, modes, various options, and control functions.

Figure 19 shows a typical screen without conversion to cylindrical coordinates, and figure 20 shows the same screen after conversion to cylindrical coordinates. Whenever you select from this menu, an asterisk appears next to the choice to aid in construction of the plot.

rho	I	CB-Numeric	Vector	Contour Options
velocity x	J	CB-Color	Contour	Vector Options
velocity y	K	B/W	Scatter	Grid Options
velocity z		Read Prior CB	Shaded	Carpet Options
total energy	Title	Save Current CB	Carpet	X_Y_Plot Options
Mag of Velocity	X-Label		X-Y	Axes On
	Y-Label	Real-Time	Overlay	Axes Off
	Z-Label	Single Mesh		Sleep Time
		Multiple Mesh		Geometry
		Reset	Help	
	Return		Next	Bind

Figure 19.—Display Specifications menu screen without converted coordinates.

4.6.1 Variable selection.— The variables available for plotting are listed in the far left column in figures 19 and 20. Position the cursor over the desired variable name and click the left mouse button.

4.6.2 Plane selection.— The planes available for plotting are listed in the second column from the left in figures 19 and 20. In figure 19, the individual indices (i, j, k) are available for specification. Position the cursor over the desired index and click the left mouse button. IDDS will now display the minimum and maximum values for the index, and prompt you to input the desired starting index, the ending index, and the index skip factor. Pressing <Return> at a prompt will give you the default value for that quantity. (Default values are the minimum index value (1), the maximum index value, and an index skip factor of one.) Referring to figure 3, if you want an ij-plane at k = 45, then select i and press <Return> three times, select j and press <Return> three times, and then select k and enter 45 <Return>, 45 <Return>, 1 <Return>. If you enter 45, 60, 5 for k, then ij-planes 45, 50, 55, and 60 will be active for plot generation. When a plot is generated, the four active planes appear on the screen together. IDDS does not yet know the planar orientation: this information will have to be supplied to IDDS when the plot type is selected.

In figure 20, the planar orientations are predetermined by IDDS as a result of the conversion process. A cross-channel orientation consists of an entire ij-plane. You cannot specify a subset of an ij-plane when viewing converted data. When you select **Cross-Channel**, you will be prompted for the k-index ranges. As before, if you enter 45, 45, 1, then only that plane is active for display. However, if you enter 45, 60, 5, then plane 45 will be displayed, followed by 50, then 55, and finally plane 60. Here, the four active planes will be displayed sequentially. A blade-to-blade orientation consists of an entire ik-plane. When you select this orientation, you will be prompted for the j-index ranges. When you select a meridional orientation, which consists of an entire jk-plane, you will be prompted for the i-index range.

rho	Cross Channel	CB-Numeric	Vector	Contour Options
velocity x	Blade to Blade	CB-Color	Contour	Vector Options
velocity y	Meridional	B/W	Scatter	Grid Options
velocity z		Read Prior CB	Shaded	Carpet Options
total energy	Title	Save Current CB	Carpet	X_Y_Plot Options
Mag of Velocity	X-Label		X-Y	Axes On
Mach Number	Y-Label	Real-Time	Overlay	Axes Off
Meridional Velocity	Z-Label	Single Mesh		Sleep Time
Quasi-Meridional Velocity		Multiple Mesh		Geometry
Spanwise Velocity				
Streamwise Velocity		Reset		Help
V_theta_prime				
U_rho	Return		Next	Bind
V_total_secondary				

Figure 20.—Display Specifications menu with converted coordinates.

4.6.3 Plot type selection. — The plot types available are listed in the fourth column from the left in figures 19 and 20. Position the cursor over the desired plot type and press the left mouse button. If the data has not been converted to cylindrical coordinates, IDDS will now prompt you for the planar orientation. The specifics of each plot will be discussed in a following section.

4.6.4 Color bar selection. — You can manipulate the color bar with the choices listed in the third column from the left. The first selection, **CB-Numeric**, lets you specify the plotting ranges. The default range is from the global minimum to the global maximum value of the variable selected for plotting.

The next choice, **CB-Color**, lets you choose a subset of the color bar. When you select **CB-Color**, a horizontal color bar is drawn on the screen, and you will be prompted to specify the lower color and then the upper color. Point the mouse at the desired color and click the left mouse button, or press <Return> and receive the default color. The default colors are the leftmost color on the screen for the lower color and the rightmost color on the screen for the upper color.

Selecting the third choice, **B/W**, allows you to plot in black and white. Here, the full-color bar is replaced with a monochrome bar ranging from black to white. The selection **B/W** on the **Display Specifications** menu will now be replaced with **Color**, allowing you to return to the standard color bar.

The remaining two choices, **Read Prior CB** and **Save Current CB**, are not yet implemented. The first is intended to let you define a color bar outside of IDDS for use in IDDS. The second will allow you to save the current color bar. This could be used when a particular subset of the color bar needs to be saved for future dataset comparisons.

4.6.5 Labeling selections. — The labeling choices are listed in the second column from the left in figures 19 and 20. Using **Title**, you can display a title of up to 80 characters in length across the top of the screen. The next three choices **X-Label**, **Y-Label**, and **Z-Table**, are for labeling the x, y, and z axes, respectively. For all of these choices, enter <Return> on a line by itself to null out an unwanted label.

4.6.6 Mode selection. — The mode selections are listed in the third column from the left. The first of these, **Real-Time**, is not yet implemented. The second choice, **Single Mesh**, is the default mode. With this choice, you can specify which mesh out of a possible 150 is to be used for plot generation. The default mesh number to be displayed is 1. The last choice, **Multiple Mesh**, allows you to specify a sequence of meshes to be displayed. This choice is connected with the first line in a header file.

4.6.7 Display options. — The display options are listed in the last column on the right. With these options, you can customize a particular plot or display. Each option, except the **X_Y** plot option, has a default setting associated with it.

4.6.7.1 Contour options: **Contour Options** allows you to set the number of contours and the contour values to desired levels. The default number of contours is 20, with the beginning level being the variable minimum, the ending level being the variable maximum, and the increment between levels being the difference of maximum value minus minimum value divided by number of contours. These default values are set once a variable is selected for plot generation.

4.6.7.2 Vector options: The second display option, **Vector Options**, is for fine tuning a vector plot. A unit vector is defined and linearly scaled according to the global range of minimum and maximum velocities. This option lets you adjust the global minimum and maximum velocities. Without this option, it was found that the secondary velocity vectors would get overwhelmed by the global range and appear as dots on the screen. The color of the vector is determined by the variable selected for display and not the magnitude of velocity. Thus, the vector orientation and size are determined by the velocity components and magnitude, respectively, and the vectors can be colored by Mach number, magnitude, pressure, or whatever you desire. The range of the color scale is determined by the color bar options.

4.6.7.3 Grid options: Use **Grid Options** to specify the color of the grid.

4.6.7.4 Carpet plot options: **Carpet Plot Options**, listed next, allows you to specify a subset of the grid for plot generation or to reset it back to the full grid. By using this, you can zero in on a particular region of interest in the solution.

4.6.7.5 X-Y plot options: Using **X-Y Plot Options** is exactly the same as when an **X_Y** plot is first requested. You will be prompted for which grid line (or subset of) to plot. For example, if the data is converted and the active plane is $j = 2$, then you would be prompted for the i -index values and the k -index values. To see how the variable values range over the grid line $i = 2$, enter 2 for i -start and i -end, and then <Return> twice for the entire k index values. If the index values for both directions are not equal, that is, i -start $\neq i$ -end and k -start $\neq k$ -end, then the action of this routine will be to plot everything.

4.6.7.6 Axis options: **Axes On** and **Axes Off** turn axis generation on or off, respectively.

4.6.7.7 Delay time option: The next option, **Sleep Time**, lets you specify the time between displays. This is useful when stepping through the dataset from front to back (i.e., plot all cross-channel planes from $k = 1$ to $k = k_{\text{max}}$). The default value for the sleep time is zero.

4.6.7.8 Geometry option: The last option, **Geometry**, is active only if you have specified a geometry file (fig. 3). Selecting this option gives you another menu wherein the items specified in the geometry file are available for manipulation. The **Geometry** menu will be discussed in the next section.

4.6.8 Control options. — The Display Specifications menu of IDDS has five control options listed in two rows across the lower half of the screen: **Bind**, **Reset**, **Help**, **Return**, and **Next**. **Help** and **Return** function in the same manner as the other **Help** and **Return** options in IDDS.

Reset will take you back to where you first entered the **Display Specifications** menu. All options entered to that point will be lost. This is a good way to start over in case something does not appear to be working properly.

Bind tells IDDS that the specifications are finished for the window being worked on. IDDS will then check to see if there is enough information specified to produce a plot. If so, IDDS will then activate the next window for specification purposes. When you select **Bind** twice, the information contained for the first window is copied into the information area for the second window. If there is not enough information to proceed, IDDS will tell you what else you must specify for a plot to be produced. Once you have specified all windows correctly, IDDS will activate **Next**.

Next takes you to the **Display Control** menu. If you select this option before all windows are bound, you will receive an error message.

If a message comes on the screen concerning either the **Next** or **Bind** options, press <Return> to tell IDDS that you have seen the message and are ready to proceed.

4.7 Geometry Menu

You will enter this menu when you press **Geometry** in the **Display Specifications** menu. An example of this menu is shown in figure 21. The names in the figure are based on the descriptive file of figure 4.

Hub	Make Grid
Pressure	Make Surf
Suction	Grid Density
	Set Color
	Screen Color
	Display Objects
	Return
	Reset

Figure 21.—Geometry menu.

The items listed in the left column of figure 21 are directly affected by the options in the right column, which work in the following manner.

- (1) **Make Grid** is the default for drawing an object. Whenever an object is selected, it will be drawn in a grid outline.
- (2) **Make Surf** will cause all objects selected after it to be drawn solid.
- (3) By using **Grid Density**, you can have the grid drawn coarse (default) or fine. If you select a fine grid, IDDS will draw every possible grid line.
- (4) **Set Color** sets the drawing color. When you select this option, the screen will contain eight possible colors to choose from. Place the cursor over the desired color and click the left mouse button.
- (5) **Screen Color** sets the screen color. Like the **Set Color** option, the screen contains eight possible screen color selections. Note that the screen color can be selected only when displaying the geometry: this color choice does not carry over when displaying the data. The default screen color is black.
- (6) **Display Objects** causes all objects selected to be displayed.
- (7) **Return** takes the user back to the **Display Specifications** menu.
- (8) **Reset** will null out all options set to that point. In effect, you will begin again with a clean slate.

When you select an object, **Hub**, for example, IDDS will prompt you for the number of “Hub” to display, and then for a “Hub” skip factor. The skip factor refers to how many empty slots to leave between each replica of “Hub.” For example, if you tell IDDS that you would like to see two copies of “Hub,” and you enter a skip factor of 1, then one hub would be drawn according to the grid coordinates, $2 \times \text{PI} / \text{Total_Number_of_Blades}$ would be skipped, and another “Hub” would be drawn (fig. 4, line 2). Thus, to draw out the entire hub of the machine, enter the total number of blades for the number of “Hub” to be drawn and enter zero for the skip factor.

4.8 Display Control Menu

When you select **Next** from the **Display Specifications** menu, you will enter the **Display Control** menu. There are seven items in this menu, as shown in figure 22. By selecting **Draw Grid**, **Draw Plot**, or **Draw Geometry**, you are telling IDDS what to display. You can select and display these items separately or you can combine them in any manner.

Draw Plot	Draw Grid	Draw Geometry
Reset	Help	Return
Produce Display		

Figure 22.—Display Control menu.

Reset clears all instructions for what to display. That is, individual plot, grid, and geometry characteristics are left alone, but the internal codes to display the plot, the grid, or the geometry are cleared.

Produce Display initiates the plotting sequence. All choices made in this menu will be displayed.

Return takes you back to the **Display Specifications** menu.

5.0 PLOTTING CAPABILITIES

This section discusses the various plots that IDDS can produce. There are five different plot types to choose from the **Display Specifications** menu: **Contour**, **Vector**, **Shaded**, **X-Y**, and **Carpenter**. All plots are available with either two-dimensional or three-dimensional data. If the dataset to be displayed is 3 dimensional and it has been converted to cylindrical coordinates, then all plots with that data will be mapped to a two-dimensional space and displayed. If the geometry is to be displayed along with a converted plot, then the plot will be displayed in its natural coordinate system (i.e., three dimensional). IDDS automatically switches back and forth between two-dimensional and three-dimensional plot generation depending on the display of geometry.

When the data have been converted to cylindrical coordinates, IDDS will automatically combine indices and predetermine three views. These views are cross-channel, blade-to-blade, and meridional. The cross-channel view is formed by combining all i- and j-indices. This is intended to show secondary flow characteristics. A blade-to-blade view is developed by combining all i- and k-indices. This view is intended to show streamwise flow characteristics on a plane parallel to the hub from inlet to exit. The meridional view is constructed by combining all j- and k-indices. This view is also intended to show streamwise flow characteristics but in a plane parallel to a blade surface from inlet to exit.

5.1 Contour Plots

When you select a contour plot, IDDS automatically chooses the beginning contour level, the ending contour level, the increment between contour levels, and the number of contour levels. The contour levels are based on the global minimum and maximum for the variable in question. If you wish, you can override these values and have IDDS plot specific contour levels. You can also change the number of contour levels that IDDS will plot, the default being 20 contour levels. Make these changes via **Contour Options** in the **Display Specifications** menu.

5.2 Vector Plots

When you select a vector plot, IDDS will automatically use the minimum and maximum vector magnitudes as a scale for vector sizing. The color of the vector is determined by the variable selected from the first column of the **Display Specifications** menu. Thus, a vector plot shows magnitude and direction along with how a variable varies in the vector field.

When a dataset has been converted to cylindrical coordinates, the vector plots are displayed in two dimensions. A cross-channel view shows the secondary flow characteristics, and both blade-to-blade and meridional views show the streamwise flow characteristics.

5.3 Shaded Plots

A shaded display is simply a filled-in contour plot. Any variable can be used for plotting purposes. You can control the plotting range of the variable via the color bar limits and adjust the color range through the color bar range selection.

5.4 X-Y Plots

An X-Y plot consists of a set of axes with grid indices along the X-axis and variable values along the Y-axis. The grid indices should vary in one direction only. For example, if a grid is 44 by 45 by 106 (i, j, k) and you want to see how a variable varies across i at j = 2 and k = 55, then when prompted for i-index values, enter <Return>, <Return>, <Return>, 2, 2, 1 for j, and 55, 55, 1 for k. If the data have been converted to cylindrical coordinates, you should be mindful of the axial orientations for cross-channel, blade-to-blade, and meridional plots. For example, if you select a blade-to-blade view, then an ik-plane is active for plotting. You would then tell IDDS which j-index to plot. Thus, when you select the X-Y plot, you must choose which grid line to plot, either an i or k grid line. If you enter <Return> at both the i and k prompts, then every value on the j-plane selected will be plotted.

5.5 Carpet Plots

A carpet plot takes a plane of data and deforms the plane according to variable values in the third dimension. The variable values at the grid nodes determine the color and height of the resulting grid. If a nodal value is above or below the maximum or minimum, respectively, then the height is clamped to the respective level.

When using cylindrical coordinates, IDDS can determine the planar orientation exactly. However, when using nonconverted coordinates, you must inform IDDS of the planar orientation via a menu selection.

6.0 EXAMPLE SESSION

This chapter presents an example session of using IDDS at a workstation, together with some samples of plots drawn with IDDS. This example assumes that the data files to be used are located on the machine containing IDDS. If the files are stored elsewhere, you must issue the **nfsmount** command prior to invoking IDDS. Any path name to the file must precede the filename when entered. The following pages will show a sample screen accompanied by a menu system diagram. The selections you are required to make will be underlined, and the path you follow through the menu system will be highlighted in the diagram. Appendix B gives a quick guide to using IDDS and a list of hot keys for manipulating the plots.

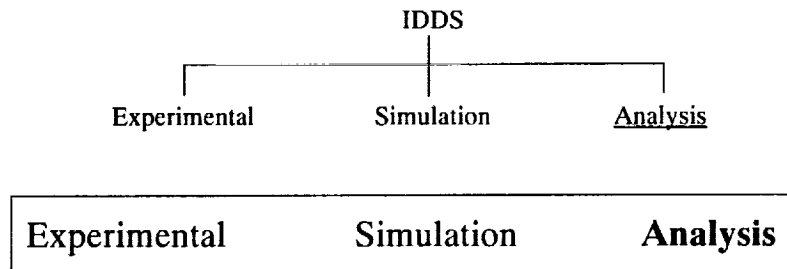


Figure 23.—Selecting Analysis from the global menu.

In the first example screen layout (fig. 23), position the cursor over **Analysis** and click the left mouse button. The other two selections, **Simulation** and **Experimental**, are undergoing development and will be implemented at a later date.

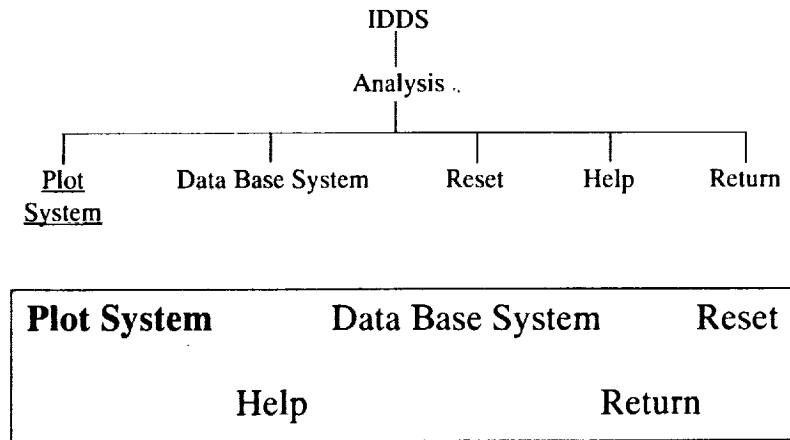


Figure 24.—Selecting Plot System from the Analysis subsystem main menu.

In the next example screen layout (fig. 24), position the cursor over **Plot System** and click the left mouse button. This will activate the plotting routines and their associated menu structure in a stand-alone operating mode.

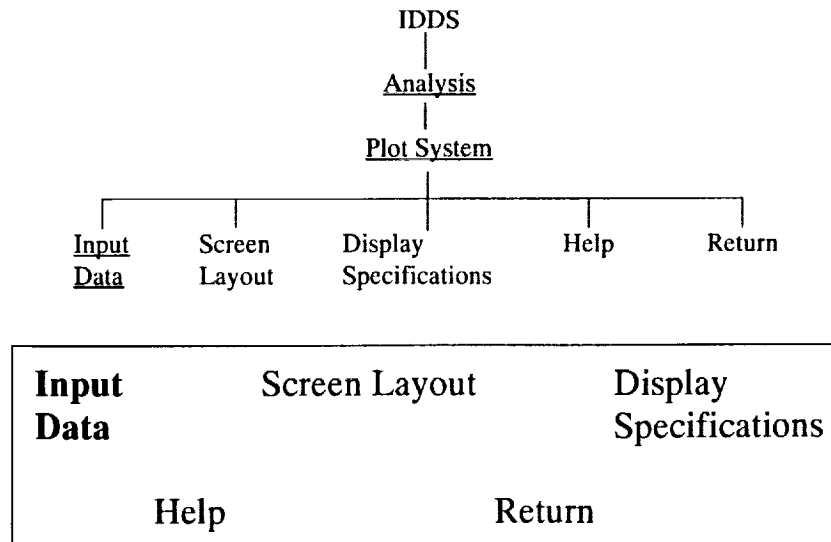


Figure 25.—Selecting Input Data from the IDDS main menu.

Here, select **Input Data** to start the data specification process (see fig. 25). You are now in the first of four main submenus of IDDS.

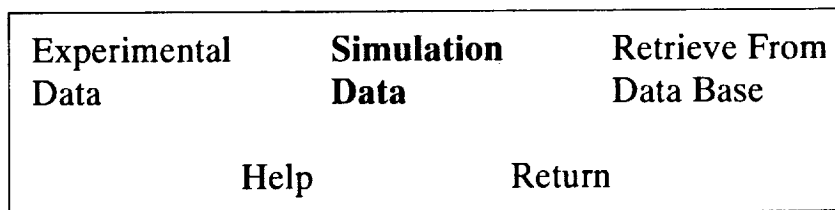
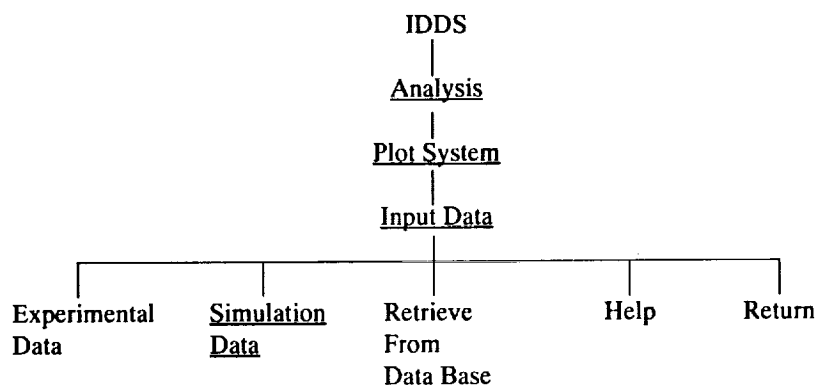


Figure 26.—Selecting Simulation Data from the Data Input menu.

Next, position the cursor over **Simulation Data** (fig. 26) and click the left mouse button. This tells IDDS that the incoming data will be from a simulation code. The distinction between simulation and experimental datasets is made so that comparisons of the two can be conducted in the future.

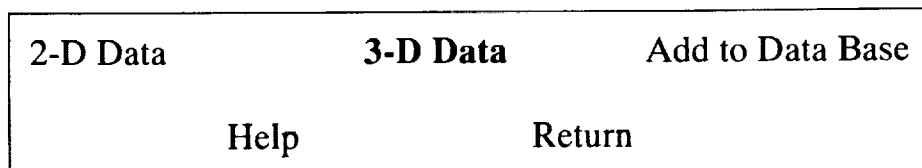
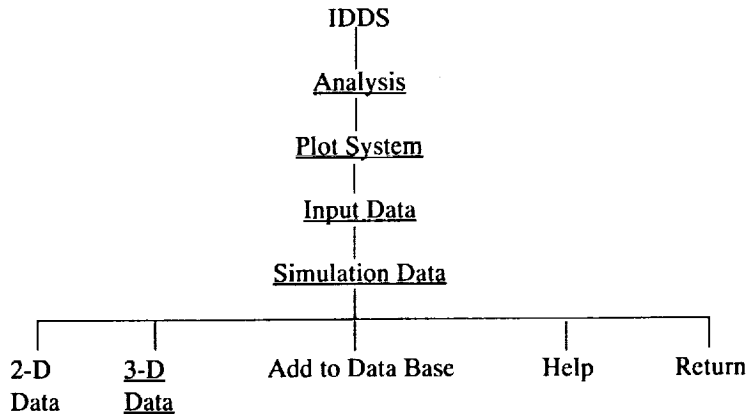
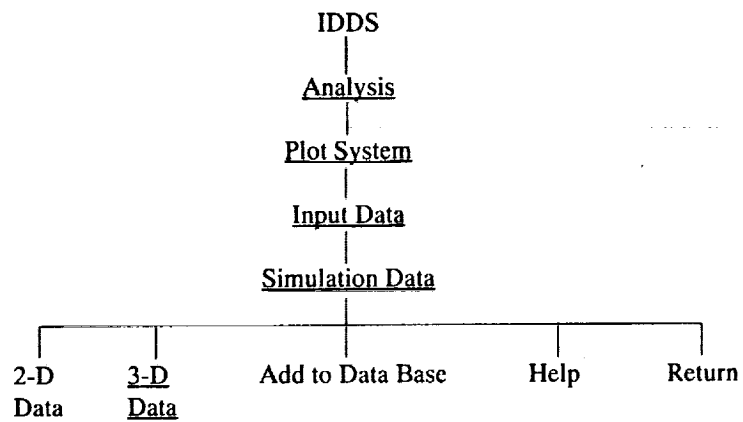


Figure 27.—Selecting 3-D Data from the Simulation Data menu.

Now, as shown in figure 27, position the cursor over **Three-Dimensional Data** and click the left mouse button. This action and the previous selection, **Simulation Data**, tell IDDS that the data to be read in is a three-dimensional simulation dataset.



Enter fluids file name:

Enter grid file name:

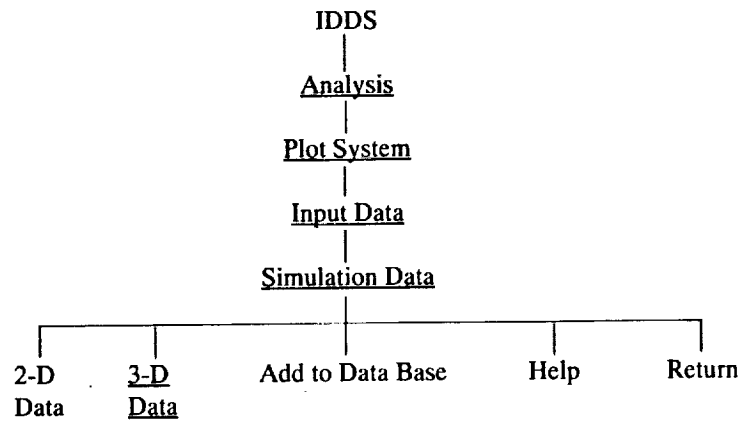
Enter geometry file name:

Enter header file name:

Convert to cylindrical coordinates? <N>

Figure 28.—Data input prompts.

Figure 28 shows the prompts you must answer for the input of data files.



Please enter the following values:

Omega (rpm) <0>

Plenum Temperature <0>

Total Pressure <0>

Cp <0>

Figure 29.—Conversion prompts.

Figure 29 shows the prompts which you must answer for the conversion of data files. You are responsible for assuring unit consistency between these values and those associated with the data file. The default values are shown between the <>.

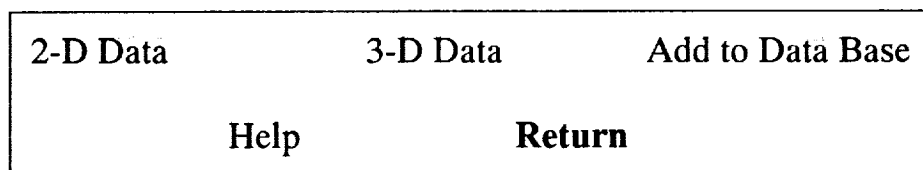
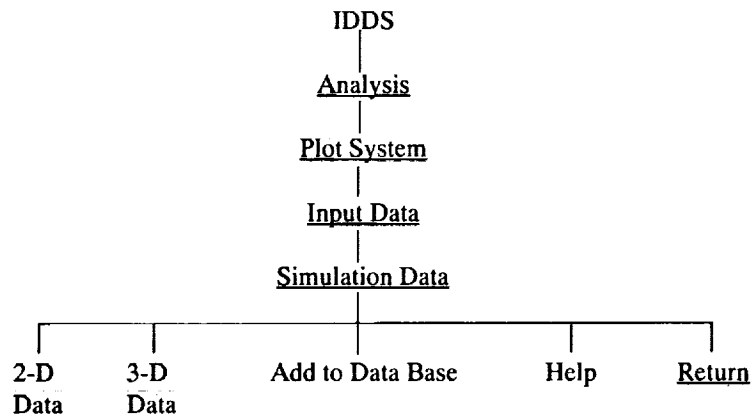


Figure 30.—Backing out of the Simulation Data menu.

Now, select **Return** from the Simulation Data menu (fig. 30) and start moving back up the menu hierarchy.

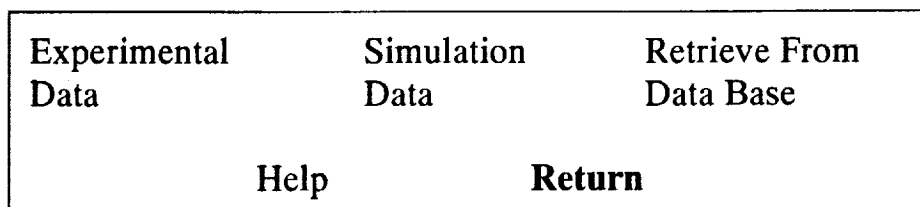
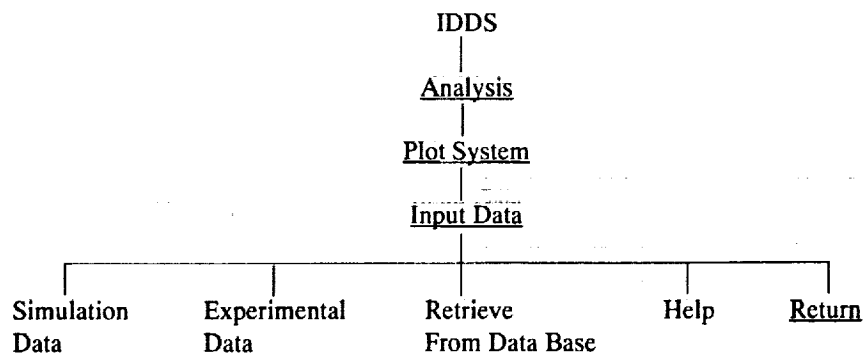


Figure 31.—Backing out of the Input Data menu.

Select **Return** again (see fig. 31) and continue moving back up the menu hierarchy.

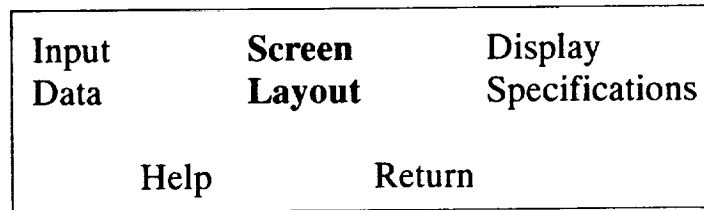
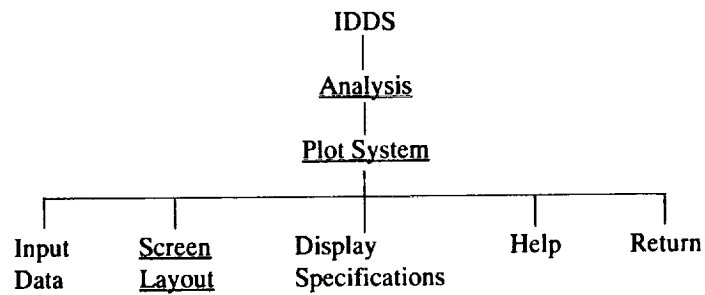


Figure 32.—Selecting Screen Layout from the IDDS main menu.

From the IDDS main menu (fig. 32), select **Screen Layout** to configure the screen. You are now in the second of four main sub-menus.

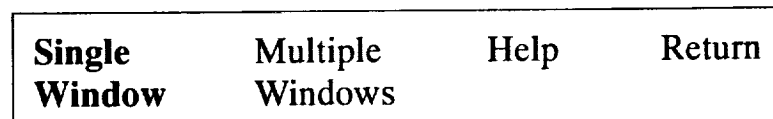
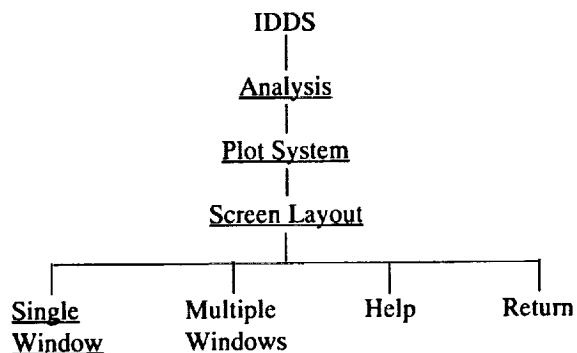
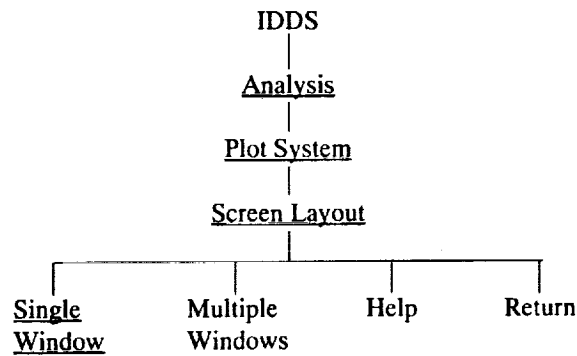


Figure 33.—Selecting Single Window from the Screen Layout menu.

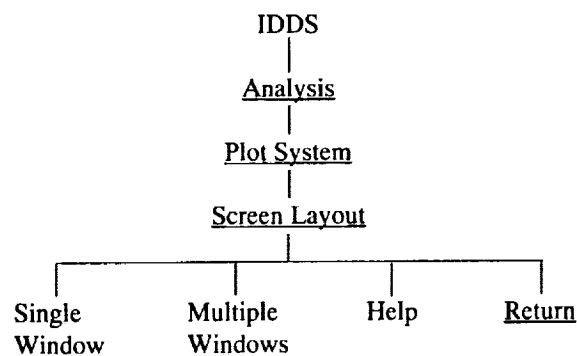
Select **Single Window** (fig. 33). Now the entire screen can be used for display purposes. Once you make this selection, you will be asked to associate a window with a file. This is covered on the next page.



*	Single	*	File 1
	Reset Window		Bind
	Reset File		Return
	Reset All		Help

Figure 34.—Selecting from the Window and File Association menu.

Figure 34 shows the Window and File Association menu. Now select the file input and **Single**, then **Bind** and, finally, **Return**. See section 3.5 for a more thorough explanation of the control options listed here. The only way to enter this menu is when you have specified at least one file and you have requested a unique screen layout.



Single Window	Multiple Windows	Help	Return
---------------	------------------	------	---------------

Figure 35.—Backing out of the Screen Layout menu.

Once the data have been read in and, if necessary, converted, you will go back to the **Screen Layout** menu. You can change screen configurations at this point. For this example, select **Return** (fig. 35) to go up one level in the menu system.

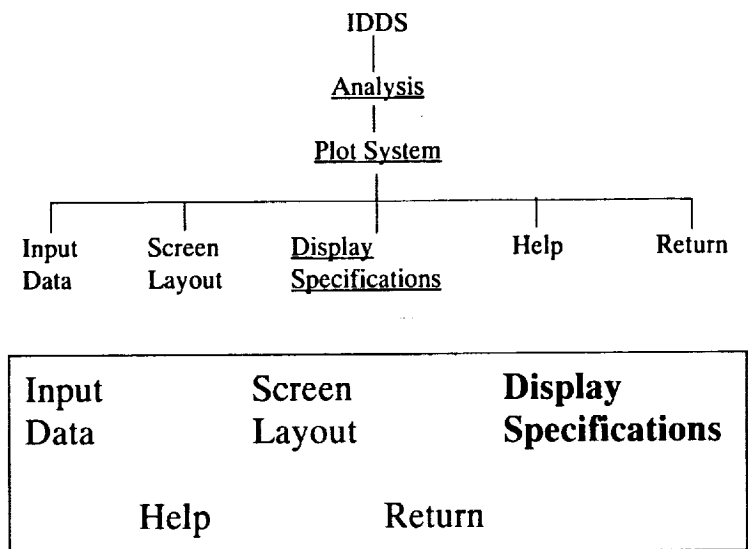


Figure 36.—Selecting Display Specifications from the IDDS main menu.

Selecting **Display Specifications** from the IDDS main menu (fig. 36) will now allow you to specify a plot. See section 3.6 for a more detailed discussion of the options in this menu. You must select three things: (1) a variable, (2) a planar orientation, and (3) a plot type. Once these have been selected, you must select **Bind** and then **Next**.

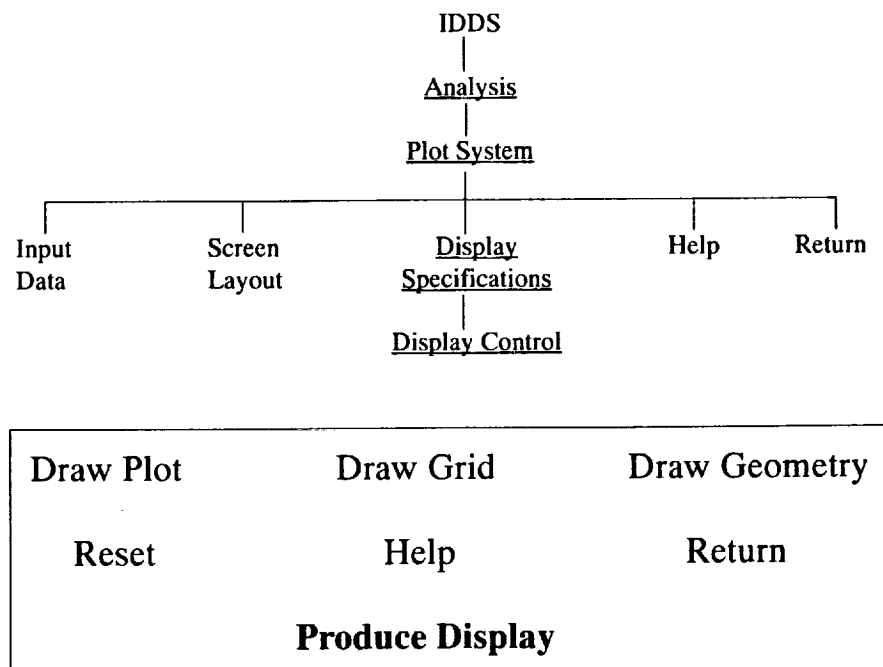


Figure 37.—Display Control menu.

From the **Display Control** menu (fig. 37), you can display the plot, the grid, or the geometry, by themselves or in any combination. Note that the geometry can be displayed only if you have entered a geometric file and if you entered the **Geometry** submenu from the **Display Specifications** menu.

The following figures depict typical plots from IDDS. All plots are of the flow in a centrifugal compressor equipped with a splitter vane. The overall grid dimensions are 45 by 44 by 106 ((i, j, k), (x, y, z)). The data have been converted to cylindrical coordinates.

(1) Figure 38 shows a contour plot of the magnitude of velocity in a meridional (jk) plane. This particular plane has an i-index of 3.

(2) Figure 39 shows the same data as figure 38 using a shaded display plot.

(3) Figure 40 is a shaded display of the magnitude of velocity in a cross-channel (ij) plane at $k = 78$. The central space between the two halves of plot depicts the main blade, whereas the solid purple in each half shows the splitter vane.

(4) Figure 41 is a vector plot of the secondary velocity field in the same cross-channel plane as figure 40. The vectors are colored by magnitude of velocity, and their length is determined by the relation of the secondary velocity to the overall global magnitude of velocity.

(5) Figure 42 shows the magnitude of velocity in a blade-to-blade (ik) plane near the hub ($j = 2$).

(6) Figure 43 illustrates a contour plot of the magnitude of velocity in a blade-to-blade orientation. This plot was constructed by telling IDDS to display the hub geometry and the defined plot. Note the highly curved nature of the geometry.

(7) Figure 44 is a carpet plot of a cross-channel plane at $k = 60$.

(8) Figure 45 shows an X-Y plot based on a cross-channel plane of $k = 78$. The plot spans i (i.e., i ranges from 1 to 45) and is held constant at $j = 2$.

(9) Figure 46 shows the magnitude of velocity plotted on the entire centrifugal compressor geometry. The view is head-on down the axis of rotation. White gaps in the plot indicate main blades, and the solid purple areas correspond to the splitter vanes.

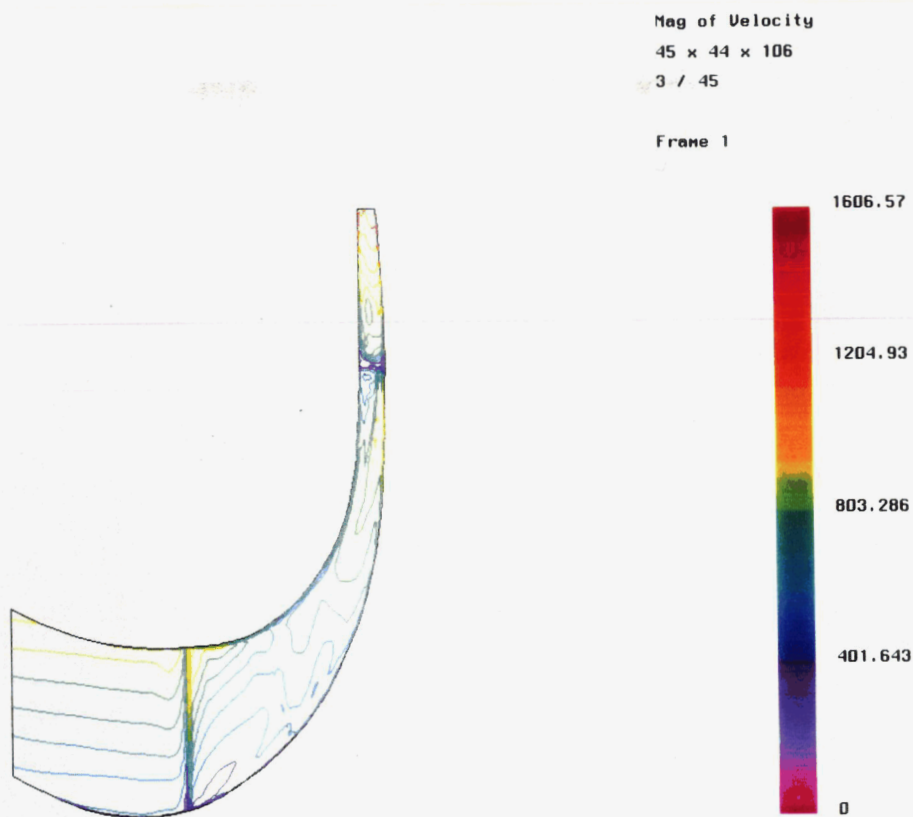


Figure 38.—Meridional plane contour plot.

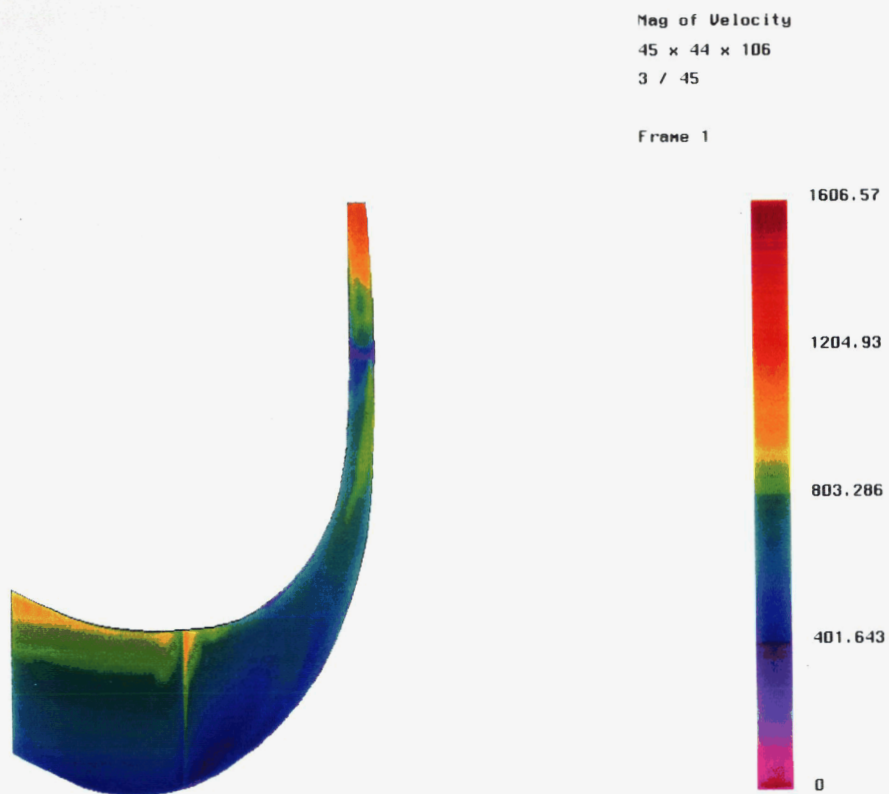


Figure 39.—Meridional plane shaded display.

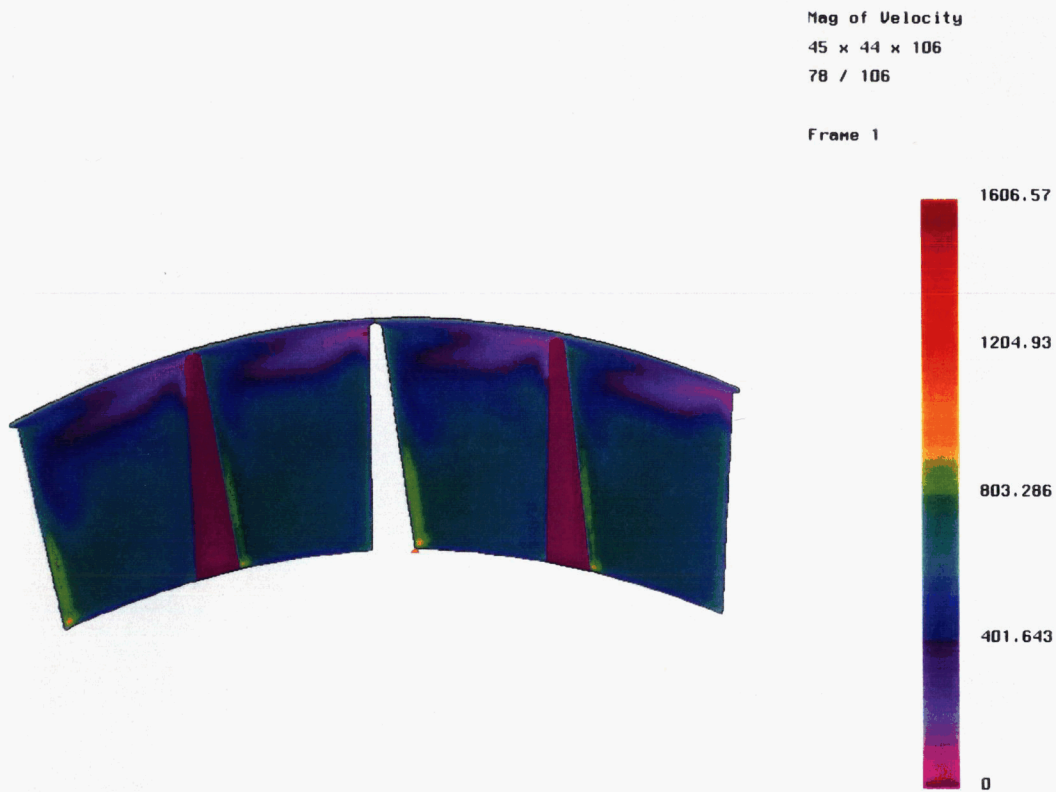


Figure 40.—Cross-channel plane shaded display.

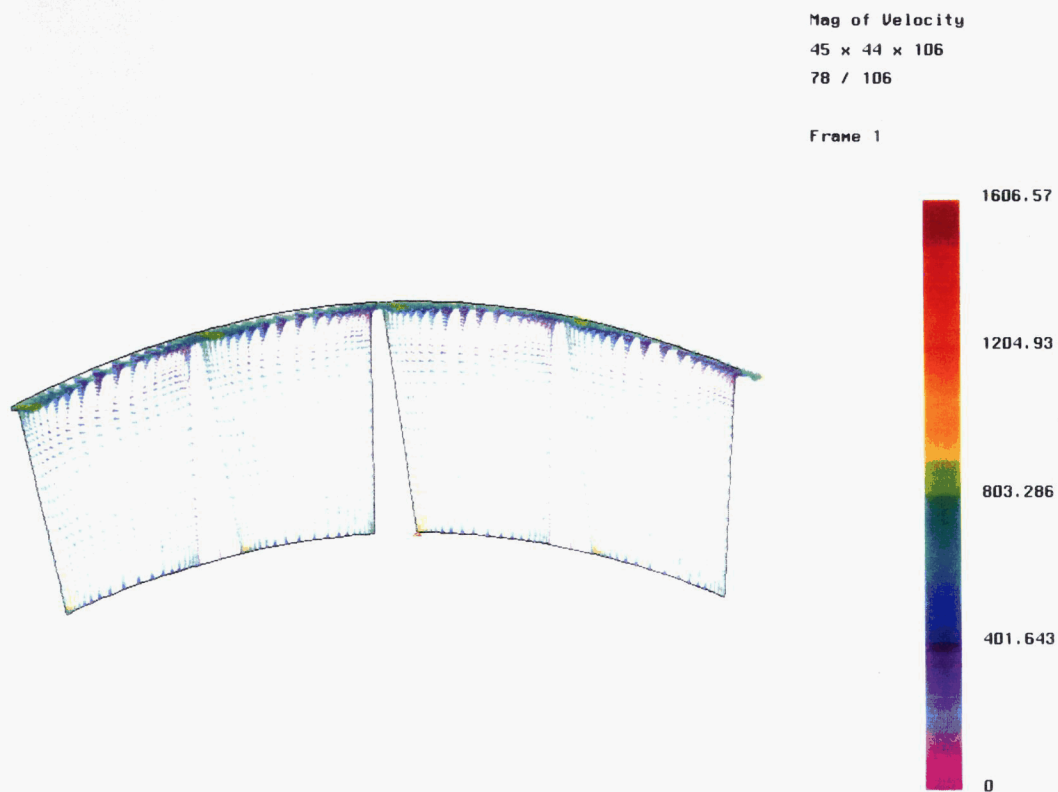


Figure 41.—Cross-channel plane secondary velocity field.



Figure 42.—Blade-to-blade plane shaded display.



Figure 43.—Blade-to-blade plane contour plot.

28 INTENTIONALLY BLANK

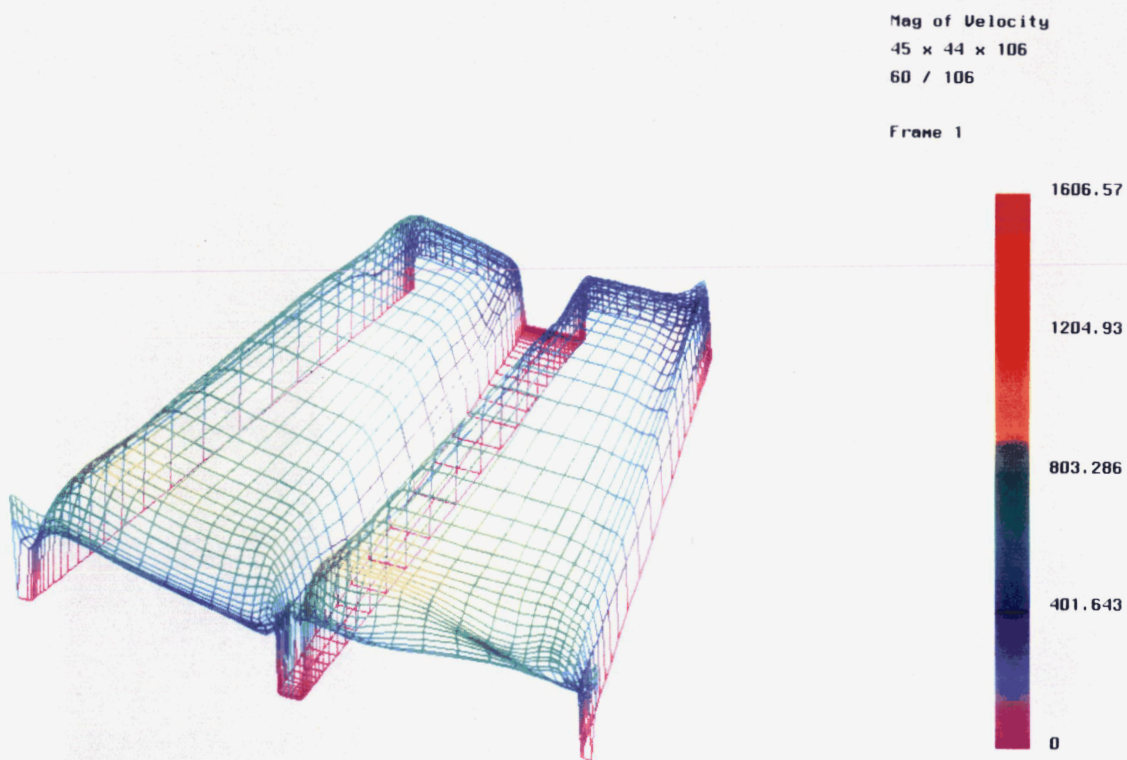


Figure 44.—Cross-channel plane carpet plot.

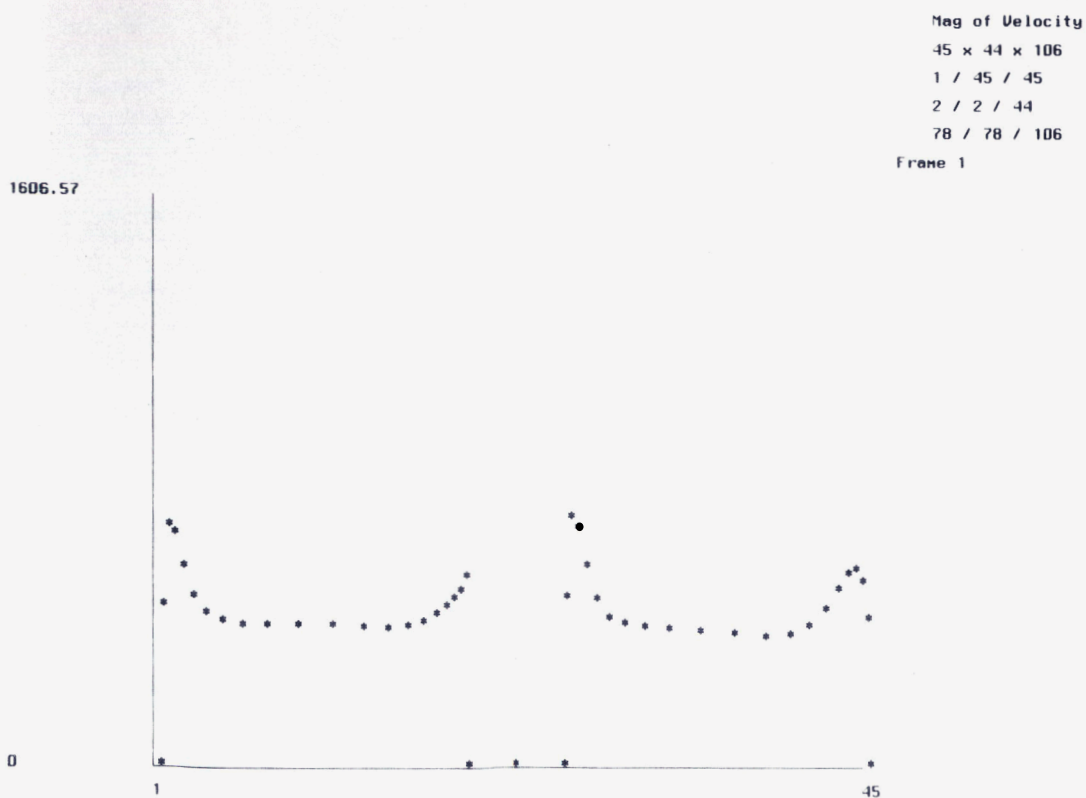


Figure 45.—x-y plot.



Figure 46.—Entire flow-field.

ORIGINAL PAGE
 COLOR PHOTOGRAPH

APPENDIX A

EXAMPLE CODES FOR GENERATION OF DATA FILES

This appendix contains various code segments which illustrate how to properly format the data and grid files for IDDS. Examples 1 through 10 are written in the C programming language, and examples 11 through 20 are written in the Fortran programming language. The correct format is that of PLOT3D. Any PLOT3D file can be input directly to IDDS.

Examples 1 and 11 show how to write out a two-dimensional dataset. Examples 2 and 12 show how to write out a three-dimensional dataset. Examples 3 and 13 illustrate how to write out a two-dimensional dataset with more than one time-step, and examples 4 and 14 also illustrates a multiple time-step dataset but for three-dimensional data. Note that the grid dimensions in examples 3, 4, 13, and 14 could change. The next examples deal with writing out grid files. Examples 5 and 15 deal with a single two-dimensional grid. Examples 6 and 16 illustrate a multiple time-step two-dimensional grid. Note that the grid dimensions could change from time-step to time-step. Examples 7 and 17 show multiple two-dimensional grids with the same grid. Examples 8, 9, 10, 18, 19 and 20 deal with the same issues as examples 5, 6, 7, 15, 16 and 17 but in a three-dimensional fashion. Table I summarizes this information. In all examples, it is assumed that the i-index corresponds to X; the j-index corresponds to Y; and the k-index corresponds to Z. These conventions are shown in Figure 47.

TABLE I.—EXAMPLE CODE SEGMENTS

Programming language		File type
C	FORTTRAN	
1	11	Single Mesh Two-dimensional Data
2	12	Multiple Mesh Two-dimensional Data
3	13	Single Mesh Three-dimensional Data
4	14	Multiple Mesh Three-dimensional Data
5	15	Single Mesh Two-dimensional Grid
6	16	Multiple Mesh Two-dimensional Grid
7	17	Multiple Mesh Two-dimensional Grid Same Grid Size
8	18	Single Mesh Three-dimensional Grid
9	19	Multiple Mesh Three-dimensional Grid
10	20	Multiple Mesh Three-dimensional Grid Same Grid Size

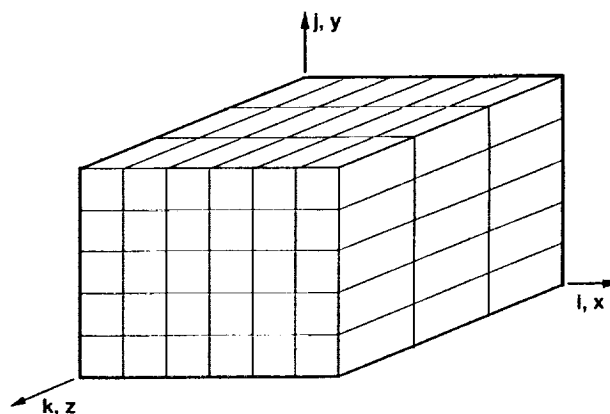


Figure 47.—Index conventions.

Example 1 – Two-Dimensional Dataset (C Language)

```
fwrite(&i_max, 4, 1, data_file);
fwrite(&j_max, 4, 1, data_file);

fwrite(&amach, 4, 1, data_file);
fwrite(&alpha, 4, 1, data_file);
fwrite(&Reynolds, 4, 1, data_file);
fwrite(&time, 4, 1, data_file);

for(variable_num = 0; variable_num < 5; variable_num++)
    for(j = 0; j < j_max; j++)
        for(i = 0; i < i_max; i++)
            fwrite(&data[variable_num][i][j], 4, 1, data_file);
```

Example 2 – Three-Dimensional Dataset (C Language)

```
fwrite(&i_max, 4, 1, data_file);
fwrite(&j_max, 4, 1, data_file);
fwrite(&k_max, 4, 1, data_file);

fwrite(&amach, 4, 1, data_file);
fwrite(&alpha, 4, 1, data_file);
fwrite(&Reynolds, 4, 1, data_file);
fwrite(&time, 4, 1, data_file);

for(variable_num = 0; variable_num < 5; variable_num++)
    for(k = 0; k < k_max; k++)
        for(j = 0; j < j_max; j++)
            for(i = 0; i < i_max; i++)
                fwrite(&data[variable_num][i][j][k], 4, 1, data_file);
```

Example 3 – Two-Dimensional Dataset, Multiple Time-Steps (C Language)

```
for(n = 0; n < max_number_time_steps; n++)
{
    fwrite(&i_max[n], 4, 1, data_file);
    fwrite(&j_max[n], 4, 1, data_file);

    fwrite(&amach, 4, 1, data_file);
    fwrite(&alpha, 4, 1, data_file);
    fwrite(&Reynolds, 4, 1, data_file);
    fwrite(&time, 4, 1, data_file);

    for(variable_num = 0; variable_num < 5; variable_num++)
        for(j = 0; j < j_max[n]; j++)
            for(i = 0; i < i_max[n]; i++)
                fwrite(&data[n][variable_num][i][j], 4, 1, data_file);
}
```


Example 4 – Three-Dimensional Dataset, Multiple Time-Steps (C Language)

```
for(n = 0; n < max_number_time_steps; n++)
{
    fwrite(&i_max[n], 4, 1, data_file);
    fwrite(&j_max[n], 4, 1, data_file);
    fwrite(&k_max[n], 4, 1, data_file);

    fwrite(&amach, 4, 1, data_file);
    fwrite(&alpha, 4, 1, data_file);
    fwrite(&Reynolds, 4, 1, data_file);
    fwrite(&time, 4, 1, data_file);

    for(variable_num = 0; variable_num < 5; variable_num++)
        for(k = 0; k < k_max[n]; k++)
            for(j = 0; j < j_max[n]; j++)
                for(i = 0; i < i_max[n]; i++)
                    fwrite(&data[n][variable_num][i][j][k], 4, 1, data_file);
}
```

Example 5 – Two-Dimensional Grid File, Single Time-Step (C Language)

```
fwrite(&i_max, 4, 1, grid_file);
fwrite(&j_max, 4, 1, grid_file);

for(j = 0; j < j_max; j++)
    for(i = 0; i < i_max; i++)
        fwrite(&x[i][j], 4, 1, grid_file);

for(j = 0; j < j_max; j++)
    for(i = 0; i < i_max; i++)
        fwrite(&y[i][j], 4, 1, grid_file);
```

Example 6 – Two-Dimensional Grid File, Multiple Time-Steps, Changing Grid Size (C Language)

```
for(n = 0; n < max_number_time_steps; n++)
{
    fwrite(&i_max[n], 4, 1, grid_file);
    fwrite(&j_max[n], 4, 1, grid_file);

    for(j = 0; j < j_max[n]; j++)
        for(i = 0; i < i_max[n]; i++)
            fwrite(&x[n][i][j], 4, 1, grid_file);

    for(j = 0; j < j_max[n]; j++)
        for(i = 0; i < i_max[n]; i++)
            fwrite(&y[n][i][j], 4, 1, grid_file);
}
```

Example 7 – Two-Dimensional Grid File, Multiple Time-Steps, Constant Grid Size, (C Language)

```
fwrite(&i_max, 4, 1, grid_file);
fwrite(&j_max, 4, 1, grid_file);

for(j = 0; j < j_max; j++)
    for(i = 0; i < i_max; i++)
        fwrite(&x[i][j], 4, 1, grid_file);

for(j = 0; j < j_max; j++)
    for(i = 0; i < i_max; i++)
        fwrite(&y[i][j], 4, 1, grid_file);

repeat = -1;
for(n = 0; n < max_number_time_steps - 1; n++)
{
    fwrite(&repeat, 4, 1, grid_file);
    fwrite(&repeat, 4, 1, grid_file);
}
```

Example 8 – Three-Dimensional Grid File, Single Time-Step (C Language)

```
fwrite(&i_max, 4, 1, grid_file);
fwrite(&j_max, 4, 1, grid_file);
fwrite(&k_max, 4, 1, grid_file);

for(k = 0; k < k_max; k++)
    for(j = 0; j < j_max; j++)
        for(i = 0; i < i_max; i++)
            fwrite(&x[i][j][k], 4, 1, grid_file);

for(k = 0; k < k_max; k++)
    for(j = 0; j < j_max; j++)
        for(i = 0; i < i_max; i++)
            fwrite(&y[i][j][k], 4, 1, grid_file);

for(k = 0; k < k_max; k++)
    for(j = 0; j < j_max; j++)
        for(i = 0; i < i_max; i++)
            fwrite(&z[i][j][k], 4, 1, grid_file);
```

Example 9 – Three-Dimensional Grid File, Multiple Time-Steps, Changing Grid Size (C Language)

```
for(n = 0; n < max_number_time_steps; n++)
{
    fwrite(&i_max[n], 4, 1, grid_file);
    fwrite(&j_max[n], 4, 1, grid_file);
    fwrite(&k_max[n], 4, 1, grid_file);

    for(k = 0; k < k_max[n]; k++)
        for(j = 0; j < j_max[n]; j++)
            for(i = 0; i < i_max[n]; i++)
                fwrite(&x[n][i][j][k], 4, 1, grid_file);

    for(k = 0; k < k_max[n]; k++)
        for(j = 0; j < j_max[n]; j++)
            for(i = 0; i < i_max[n]; i++)
                fwrite(&y[n][i][j][k], 4, 1, grid_file);

    for(k = 0; k < k_max[n]; k++)
        for(j = 0; j < j_max[n]; j++)
            for(i = 0; i < i_max[n]; i++)
                fwrite(&z[n][i][j][k], 4, 1, grid_file);
}
```

Example 10 – Three-Dimensional Grid File, Multiple Time-Steps, Constant Grid Size (C Language)

```
    fwrite(&i_max, 4, 1, grid_file);
    fwrite(&j_max, 4, 1, grid_file);
    fwrite(&k_max, 4, 1, grid_file);

    for(k = 0; k < k_max; k++)
        for(j = 0; j < j_max; j++)
            for(i = 0; i < i_max; i++)
                fwrite(&x[i][j][k], 4, 1, grid_file);

    for(k = 0; k < k_max; k++)
        for(j = 0; j < j_max; j++)
            for(i = 0; i < i_max; i++)
                fwrite(&y[i][j][k], 4, 1, grid_file);

    for(k = 0; k < k_max; k++)
        for(j = 0; j < j_max; j++)
            for(i = 0; i < i_max; i++)
                fwrite(&z[i][j][k], 4, 1, grid_file);

repeat = -1;
for(n = 0; n < max_number_time_steps - 1; n++)
{
    fwrite(&repeat, 4, 1, grid_file);
    fwrite(&repeat, 4, 1, grid_file);
    fwrite(&repeat, 4, 1, grid_file);
}
```

Example 11 – Two-Dimensional Dataset (FORTRAN)

```
write(9, *) IMAX, JMAX
write(9, *) AMACH, ALPHA, RE, TIME
write(9, *) ((ROC(i,j), i = 1, IMAX), j = 1, JMAX),
&          ((ROVXC(i,j), i = 1, IMAX), j = 1, JMAX),
&          ((ROVYC(i,j), i = 1, IMAX), j = 1, JMAX),
&          ((ROEC(i,j), i = 1, IMAX), j = 1, JMAX)
```

Example 12 – Three-Dimensional Dataset (FORTRAN)

```
write(9, *) IMAX, JMAX, KMAX
write(9, *) AMACH, ALPHA, RE, TIME
write(9, *) (((ROC(i,j,k), i=1, IMAX), j=1, JMAX), k=1, KMAX),
&          (((ROVXC(i,j,k), i= 1, IMAX), j= 1, JMAX), k= 1, KMAX),
&          (((ROVYC(i,j,k), i= 1, IMAX), j= 1, JMAX), k= 1, KMAX),
&          (((ROVZC(i,j,k), i= 1, IMAX), j= 1, JMAX), k= 1, KMAX),
&          (((ROEC(i,j,k), i= 1, IMAX), j= 1, JMAX), k= 1, KMAX)
```

Example 13 – Two-Dimensional Dataset, Multiple Time-Steps (FORTRAN)

```
do n = 1, max_num_time_steps
  IMAX = IM(n)
  JMAX = JM(n)
  write(9, *) IMAX, JMAX
  write(9, *) AMACH, ALPHA, RE, TIME
  write(9, *) ((ROC(n,i,j), i = 1, IMAX), j = 1, JMAX),
&          ((ROVXC(n,i,j), i = 1, IMAX), j = 1, JMAX),
&          ((ROVYC(n,i,j), i = 1, IMAX), j = 1, JMAX),
&          ((ROEC(n,i,j), i = 1, IMAX), j = 1, JMAX)
end do
```

Example 14 – Three-Dimensional Dataset, Multiple Time-Steps (FORTRAN)

```
do n = 1, max_num_time_steps
  IMAX = IM(n)
  JMAX = JM(n)
  KMAX = KM(n)
  write(9, *) IMAX, JMAX, KMAX
  write(9, *) AMACH, ALPHA, RE, TIME
  write(9, *) (((ROC(n,i,j,k), i=1, IMAX), j=1, JMAX), k=1, KMAX),
&          (((ROVXC(n,i,j,k), i= 1, IMAX), j= 1, JMAX), k= 1, KMAX),
&          (((ROVYC(n,i,j,k), i= 1, IMAX), j= 1, JMAX), k= 1, KMAX),
&          (((ROVZC(n,i,j,k), i= 1, IMAX), j= 1, JMAX), k= 1, KMAX),
&          (((ROEC(n,i,j,k), i= 1, IMAX), j= 1, JMAX), k= 1, KMAX)
end do
```

Example 15 – Two-Dimensional Grid File, Single Time-Step (FORTRAN)

```
write(8, *) IMAX, JMAX
write(8, *) ((XC(i,j), i = 1, IMAX), j = 1, JMAX),
&          ((YC(i,j), i = 1, IMAX), j = 1, JMAX)
```

Example 16 – Two-Dimensional Grid File, Multiple Time-Steps, Changing Grid Size (FORTRAN)

```
do n = 1, max_num_time_steps
IMAX = IM(n)
JMAX = JM(n)
write(8, *) IMAX, JMAX
write(8, *) ((XC(n,i,j), i = 1, IMAX), j = 1, JMAX),
&          ((YC(n,i,j), i = 1, IMAX), j = 1, JMAX)
end do
```

Example 17 – Two-Dimensional Grid File, Multiple Time-Steps, Constant Grid Size (FORTRAN)

```
write(8, *) IMAX, JMAX
write(8, *) ((XC(i,j), i = 1, IMAX), j = 1, JMAX),
&          ((YC(i,j), i = 1, IMAX), j = 1, JMAX)

REPEAT = -1
do n = 1, max_num_time_steps - 1
write(8, *) REPEAT, REPEAT
end do
```

Example 18 – Three-Dimensional Grid File, Single Time-Step (FORTRAN)

```
write(8, *) IMAX, JMAX, KMAX
write(8, *) (((XC(i,j,k), i=1, IMAX), j=1, JMAX), k=1, KMAX),
& (((YC(i,j,k), i= 1, IMAX), j= 1, JMAX), k= 1, KMAX),
& (((ZC(i,j,k), i= 1, IMAX), j= 1, JMAX), k= 1, KMAX)
```

Example 19 – Three-Dimensional Grid File, Multiple Time-Steps, Changing Grid Size (FORTRAN)

```
do n = 1, max_num_time_steps
IMAX = IM(n)
JMAX = JM(n)
KMAX = KM(n)
write(8, *) IMAX, JMAX, KMAX
write(8, *) (((XC(n,i,j,k), i=1, IMAX), j=1, JMAX), k=1, KMAX),
& (((YC(n,i,j,k), i= 1, IMAX), j= 1, JMAX), k= 1, KMAX),
& (((ZC(n,i,j,k), i= 1, IMAX), j= 1, JMAX), k= 1, KMAX)
end do
```

Example 20 – Three-Dimensional Grid File, Multiple Time-Steps, Constant Grid Size (FORTRAN)

```
write(8, *) IMAX, JMAX, KMAX
write(8, *)(((XC(i,j,k),i=1, IMAX),j=1, JMAX),k=1, KMAX),
& (((YC(i,j,k), i= 1, IMAX), j= 1, JMAX), k= 1, KMAX),
& (((ZC(i,j,k), i= 1, IMAX), j= 1, JMAX), k= 1, KMAX)
```

```
REPEAT = -1
do n = 1, max_num_time_steps - 1
  write(8, *) REPEAT, REPEAT, REPEAT
end do
```

APPENDIX B

QUICK GUIDE TO IDDS

- (1) Select **Analysis**.
- (2) Select **Plot System**.
- (3) Select **Input Data**.
 - (a) Select **Simulation Data**.
 - (b) Select **3-D Data**.
 - (c) Type in name of data file.
- (4) Select **Screen Layout**.
 - (a) Select **Single Window**.
 - (b) Click on **Single** and data file.
 - (c) Click on **Bind**.
 - (d) Wait awhile for the data to be read in.
- (5) Select **Display Specifications**.
 - (a) Select variable.
 - (b) Select grid orientation.
 - (c) Select plot type.
 - (d) Customize plot as desired.
 - (e) Select **Next**.
 - (1) Select **Draw Grid, Draw Plot, and/or Draw Geometry**.
 - (2) Select **Produce Display**.

(Picture now appears on screen)

Hot Keys for Manipulating the Plot

Mouse

Left Mouse - Rotates plot about X-axis

Middle Mouse - Rotates plot about Y-axis

Right Mouse - Rotates plot about Z-axis

Left and Middle - Translates plot along X-axis

Middle and Right - Translates plot along Y-axis

Left and Right - Translates plot along Z-axis

Keyboard

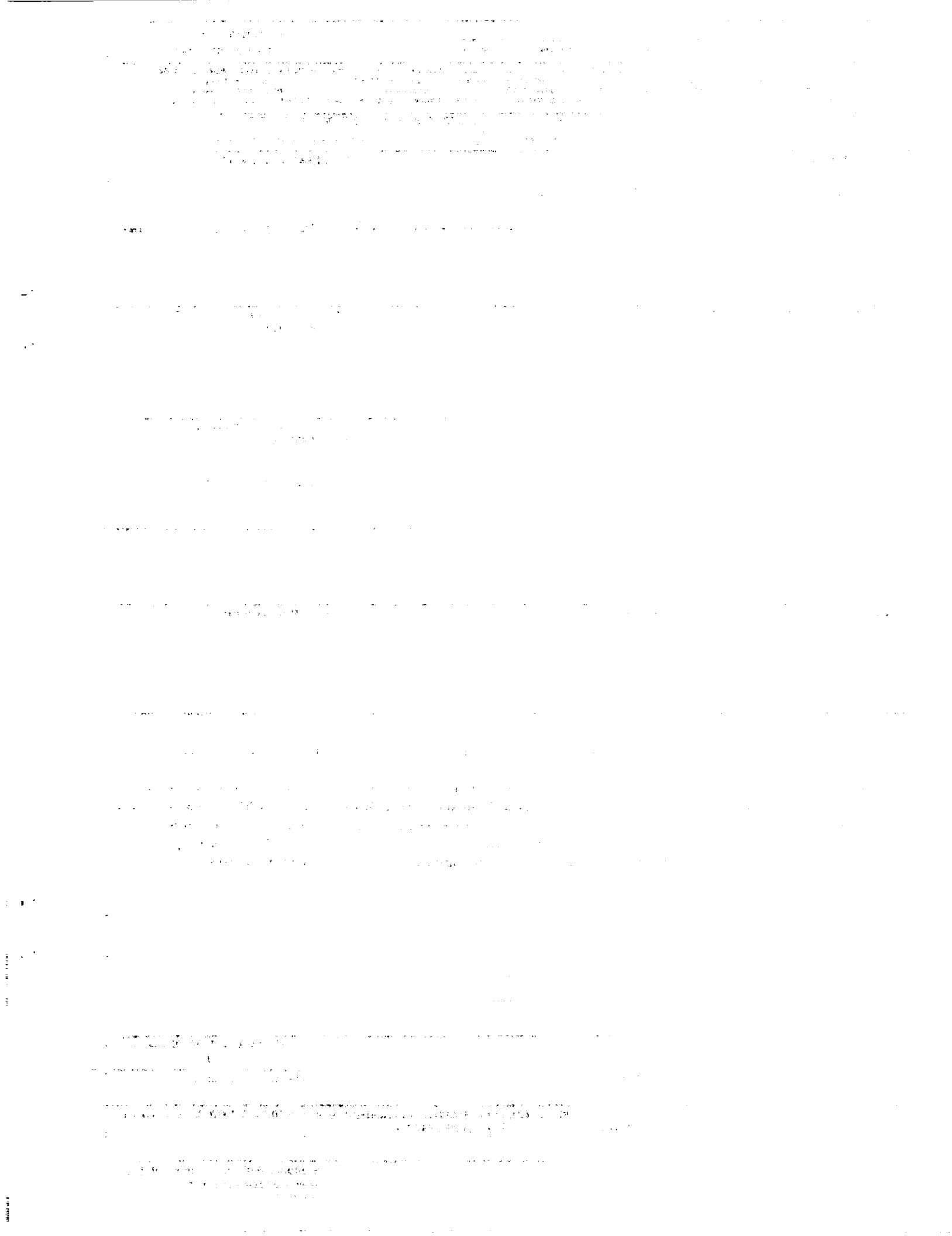
<Return> - Stops the translation or rotation process

P - Pauses the display when rendering multiple planes

R - Resumes the display after being paused

Z - Causes Z-buffering to occur, *use only when display is stopped*

<Esc> - Returns you to the display control menu (i.e., Grid, Plot, Geometry)



REPORT DOCUMENTATION PAGEForm Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE November 1992	3. REPORT TYPE AND DATES COVERED Technical Memorandum	
4. TITLE AND SUBTITLE User's Manual for Interactive Data Display System (IDDS)			5. FUNDING NUMBERS WU-505-62-52	
6. AUTHOR(S) James D. Stegeman				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) National Aeronautics and Space Administration Lewis Research Center Cleveland, Ohio 44135-3191			8. PERFORMING ORGANIZATION REPORT NUMBER E-6897	
9. SPONSORING/MONITORING AGENCY NAMES(S) AND ADDRESS(ES) National Aeronautics and Space Administration Washington, D.C. 20546-0001			10. SPONSORING/MONITORING AGENCY REPORT NUMBER NASA TM-105572	
11. SUPPLEMENTARY NOTES Responsible person, James D. Stegeman, (216) 433-3389.				
12a. DISTRIBUTION/AVAILABILITY STATEMENT Unclassified - Unlimited Subject Category 61			12b. DISTRIBUTION CODE	
13. ABSTRACT (Maximum 200 words) A computer graphics package for the visualization of three-dimensional flow in turbomachinery has been developed and tested. This graphics package, called IDDS (Interactive Data Display System), is able to "unwrap" the volumetric data cone associated with a centrifugal compressor and display the results in an easy to understand two-dimensional manner. IDDS will provide the majority of the visualization and analysis capability for the ICE (Integrated CFD and Experiment) system. This document is intended to serve as a user's manual for IDDS in a stand-alone mode. Currently, IDDS is capable of plotting two- or three-dimensional simulation data, but work is under way to expand IDDS so that experimental data can be accepted, plotted, and compared with a simulation dataset of the actual hardware being tested.				
14. SUBJECT TERMS Computer graphics; Rotating machinery; CFD			15. NUMBER OF PAGES 47	
			16. PRICE CODE A03	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT	